In the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the Application.

Listing of Claims:

1. (Currently Amended) A method for delivering an application over a network in which the business logic of the application is running on a backend server, the method comprising the steps of:

having the application invoke a GUI API to present the application's user interface;

replacing the GUI API with a re-implemented, network aware GUI API running on a backend server which translates the application's presentation layer information into pre-determined format based messages which describe a Graphical User Interface, event processing registries, and other related information corresponding to

the presentation layer of the application in high level, object level, messages;

sending such messages to the client device via a network;

processing the messages and rendering a user interface by a client-side program, which delivers a user experience for that device according to the capability of the specific client device;

rendering the user interface on the client device;

transmitting a plurality of necessary user input and a plurality of client-side events back to the server by the client-side program via a predetermined protocol;

processing the user input and client-side events on the backend server, translating the events and inputs as if they were locally generated, and sending such translated events and inputs to the application for processing;

encoding and routing output of the application to the client device using the predetermined messaging format; and,

further processing the output by the client-side program to refresh the Graphical User Interface thereat;

wherein use of the re-implemented network aware API enables the application to be developed once and deployed multiple times.

2. (Previously presented) The method of Claim 1, wherein the GUI API and the event processing API are Java Foundation Classes.

3. (Previously presented) The method of Claim 1, wherein the client-side program is a computer program based on an Operating System's API, such as Windows API, or X Windows API.

4. (Previously presented) The method of Claim 1, wherein the client-side program is a wireless device program written using the device's Operating System's API, such as Palm API and Windows CE API.

5. (Previously presented) The method of Claim 1, wherein the client-side program is a Java program written using a Java API.

6. (Previously presented) The method of Claim 5, wherein the JAVA API is AWT, Personal Java, Java 2 Micro Edition based GUI API or Java Swing.

7. (Original) The method of Claim 1, wherein the predetermined protocol is HTTP.

8. (Original) The method of Claim 1, wherein the predetermined protocol is HTTPS.

9. (Original) The method of Claim 1, wherein predetermined protocol is WAP.

10. (Original) The method of Claim 1, wherein predetermined protocol is proprietary.

11. (Previously presented) The method of Claim 1, wherein the predetermined messaging format is based on XML.

12. (Previously presented) The method of Claim 1, wherein the predetermined messaging format is proprietary.

13. (Original) The method of Claim 1, wherein the network is the Internet.

14. (Original) The method of Claim 1, wherein the network is a local area network.

15. (Original) The method of Claim 8, wherein the local area network is a bandwidth-limited slow speed network.

16. (Original) The method of Claim 1, wherein the network includes a wireless network.

17. (Previously presented) The method of Claim 11, wherein the client device is selected from the group consisting of workstations, desktops, laptops, PDAs, wireless devices and other edge devices.

18. (Original) The method of Claim 1, wherein the server and the client device are combined into one entity.

19. (Cancelled)

20. (Cancelled)

21. (Cancelled)

22. (Currently Amended) A system for distributing an application including at least a server, at least a client device, and a communication means, the system comprising:

a presentation layer of the application written using a server-side API based network programming model;

a business logic layer of the application and a data layer of the application both of which are written with the server-side API and running on the server; and where

the server-side API having a supporting infrastructure that sends the application's user interface information to a client device for presentation, handles communications problems, renders the application's user interface and dispatches necessary user input events back to the server for processing;

wherein use of the system enables the application to be developed once and deployed multiple times.

23. (Currently Amended) An apparatus for distributing an application over a network where the apparatus includes:

a server;

a client device;

a network communication means;

a re-implemented, network based API module that is used to transparently replace the API on which the application was developed;

a first means for running an application of the plurality of applications where a business logic of the application runs on the server;

a second means for replacing the API of each of the plurality of applications with the network based API so that each of the applications' logic runs on the server;

a third means for using the network based API to create a display for an application on the client device;

a fourth means for transferring the user interactions on the client device to the server, calculating the appropriate response to the input, and transmitting the appropriate response to the client machine;

a fifth means for updating the display of the application on the client device based on the responses from the server;

wherein use of the re-implemented network aware API enables the application to be developed once and deployed multiple times.

24. (Previously presented) The method of Claim 1 wherein the application code is not modified when distributing the application and the application code is not distributed to the client device.

25. (Previously presented) The method of Claim 1 used to distribute a plurality of pre-existing applications.